Journal of the Korean Physical Society, Vol. 71, No. 1, July 2017, pp. 1∼7

Meshfree Local Radial Basis Function Collocation Method with Image Nodes

Seung Ki Baek*
Department of Physics, Pukyong National University, Busan 48513, Korea

Minjae Kim
Department of Physics, Pukyong National University, Busan 48513, Korea

(Received 25 May 2017, in final form 26 May 2017)

We numerically solve two-dimensional heat diffusion problems by using a simple variant of the meshfree local radial-basis function (RBF) collocation method. The main idea is to include an additional set of sample nodes outside the problem domain, similarly to the method of images in electrostatics, to perform collocation on the domain boundaries. We can thereby take into account the temperature profile as well as its gradients specified by boundary conditions at the same time, which holds true even for a node where two or more boundaries meet with different boundary conditions. We argue that the image method is computationally efficient when combined with the local RBF collocation method, whereas the addition of image nodes becomes very costly in case of the global collocation. We apply our modified method to a benchmark test of a boundary value problem, and find that this simple modification reduces the maximum error from the analytic solution significantly. The reduction is small for an initial value problem with simpler boundary conditions. We observe increased numerical instability, which has to be compensated for by a sufficient number of sample nodes and/or more careful parameter choices for time integration.

PACS numbers: 02.60.Lj, 02.70.Jn, 05.40.Jc
Keywords: Radial basis function, Collocation, Method of images
DOI: 10.3938/jkps.71.1

I. INTRODUCTION

Numerical methods to solve a partial differential equation (PDE) are of immense importance in various branches of science and engineering, including heat transfer, structural mechanics, fluid mechanics, electromagnetism, quantum mechanics, finances, and so on. The finite-difference method (FDM) is one of the easiest to implement, but applicable to problems with relatively simple geometry. The finite-element method (FEM) allows more flexible geometry and has thus become the most widely used technique for many engineering applications. A variety of FEM packages, either commercial or non-commercial, are currently available, and they have proved the importance of numerical analysis in industries, because the method has boosted productivity by helping test prototype designs accurately.

The above methods need to decompose the problem domain into a mesh and use information of neighbors on the mesh to calculate derivatives at each given node. The construction of a mesh is often time-consuming, especially for high-dimensional complex-shaped boundary problems, and the use of the mesh becomes problematic when the object being simulated is deformed largely enough to change the connectivity between neighbors. Although we may create a new mesh during runtime, we have to assign reasonable interpolation results to the new mesh nodes based on the existing ones, which could be an additional source of error. For this reason, researchers have also devised meshfree methods, which do not require fixed connectivity between nodes. A well-known example is the Kansa method [1,2], which makes use of radial basis functions (RBF) to approximate the solution of a given PDE. This method has been successfully applied to many different problems [3–7].

One difficulty with the Kansa method is that it is not readily scalable, because one has to solve a linear system described by a fully populated $N \times N$ matrix, where N is the number of sample nodes in the domain of a given PDE. The number of operations required by a direct linear solver will be of $O(N^3)$. This is the reason that a local version of the Kansa method has been proposed in Ref. [8], because its number of required operations would then scale linearly with N. The details of the method will be given in the next section.

In this work, we show that the numerical performance of the local RBF collocation method can be improved further by a small modification, which takes into account the outside of the given domain, similarly to the

*E-mail: seungki@pknu.ac.kr; Fax: +82-51-629-5549

method of images in electrostatics [9]. In fact, the idea of using extra nodes outside the domain has already been suggested by Kansa himself in Ref. [10], and it is called 'PDE collocation on the boundary (PDECB)'. Our point is that adding extra nodes can be very costly if the complexity grows as $O(N^3)$. In the local version, on the other hand, the increment of computation would be determined by the surface-volume ratio of the system, which usually becomes negligible when we deal with a large number of sample nodes. In Sec. III, we explain our results, and compare how the results change by solving two benchmark test problems in Sec. IV. We then conclude this work in Sec. V.

II. LOCAL RBF COLLOCATION METHOD BY ŠARLER AND VERTNIK

In this section, we will explain a local version of Kansa's method in Ref. [8]. To illustrate the method, the authors of Ref. [8] have dealt with a diffusion equation

$$\rho c \frac{\partial}{\partial t} T = \nabla \cdot (k \nabla T), \tag{1}$$

where ρ, c, t, T, and k denote mass density, heat capacity, time, temperature, and thermal conductivity, respectively. The problem is defined on a spatial domain Ω with a boundary Γ. We consider three boundary conditions: Suppose a node on Γ, located at $\mathbf{r}$. The outward unit normal vector on the boundary is denoted as $\mathbf{n}$. First, the Dirichlet boundary condition fixes $T(\mathbf{r})$ to a certain value T_D. Second, the Neumann boundary condition requires that the normal derivative of T should vanish so that $\nabla T \cdot \mathbf{n} = 0$. Last, the Robin boundary condition is defined as follows:

$$\nabla T \cdot \mathbf{n} = R\left(T - T_{\mathrm{ref}}\right), \tag{2}$$

where R is a constant and T_{ref} is a reference temperature to be prescribed by the problem.

In Ref. [8], the numerical procedure to solve this PDE goes as follows:

1. Sample N_Ω nodes inside Ω and N_Γ nodes on Γ. In total, we have $N = N_\Omega + N_\Gamma$ nodes. We have chosen a regular grid for sampling the nodes to compare the results clearly, but the method works with an irregular node arrangements as well.

2. For each sample node l, determine its domain of influence ${}_l\omega$. We will focus on this particular node and its domain throughout this explanation. Let us thus drop the index l for brevity henceforth. If l lies inside Ω, ω is composed of the K nearest neighbors of l, including l itself. See Fig. 1 with $K = 5$ as an example. If l lies on Γ, it needs some care, as will be explained at the end of this section.

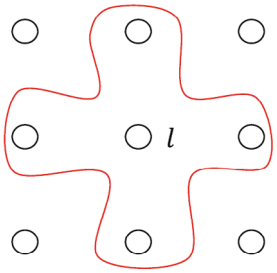

Fig. 1. (Color online) Example of the domain of influence for node l with $K = 5$ nodes, all of which lie inside Ω.

Let us denote their positions as $\mathbf{r}_n$ with $n = 1, 2, \ldots, K$. Without loss of generality, we may assign $n = 1$ to the focal node l.

3. Calculate the distance between every pair of sample nodes inside ω and define d_0 as the longest one. This parameter is used in the RBF for this ω, defined in a multiquadric form

$$\psi_k(\mathbf{r}) = \left[d_k^2(\mathbf{r}) + c^2 d_0^2\right]^{1/2}, \tag{3}$$

where d_k is the distance from $\mathbf{r}$ to node k inside ω ($k = 1, 2, \ldots, K$) and c is a shape parameter.

4. If l lies inside Ω and not on Γ, determine the collocation coefficients α_k's such that reproduce the values of T for all the K sample nodes inside ω. Specifically, we have to solve the following set of linear equations

$$T(\mathbf{r}_n) = \sum_{k=1}^{K} \psi_k(\mathbf{r}_n)\alpha_k \tag{4}$$

with $n = 1, 2, \ldots, K$. For example, if $K = 5$, the equation is written as

$$\begin{pmatrix} T(\mathbf{r}_1) \\ T(\mathbf{r}_2) \\ \vdots \\ \vdots \\ T(\mathbf{r}_5) \end{pmatrix} = \begin{pmatrix} \psi_{11} & \psi_{12} & \cdots & \psi_{15} \\ \psi_{21} & \psi_{22} & \cdots & \psi_{25} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{51} & \psi_{52} & \cdots & \psi_{55} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \vdots \\ \alpha_5 \end{pmatrix}, \tag{5}$$

where $\psi_{nk} \equiv \psi_k(\mathbf{r}_n)$. Note that Eq. (3) is readily differentiable so that we can approximate the derivatives of T in the target PDE by taking derivatives on the right-hand side of Eq. (4) once α_k's are identified. By applying an explicit time integration scheme to Eq. (1), calculate a new value of T at the focal node l. Repeat this procedure for all the N_Ω sample nodes inside Ω, and update T there.

5. Now we come to the other case that l lies on Γ. Inside its domain of influence ω, we may generally assume that K_Ω nodes are domain nodes

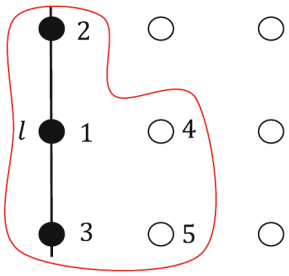

Fig. 2. (Color online) Example of ω when l lies on Γ represented by the vertical line. The empty and filled circles represent nodes in Ω and those in Γ, respectively.

whereas the other K_Γ nodes lie on boundaries, with $K = K_\Omega + K_\Gamma$. In constructing a matrix equation such as Eq. (5), we use the information on the boundary conditions for the latter K_Γ nodes. For example, suppose $K_\Gamma = 2$: We have $\partial T/\partial x = 0$ at $\mathbf{r}_1$ due to the Neumann boundary condition, and the temperature is fixed to T_D by the Dirichlet boundary condition at $\mathbf{r}_2$. We thus obtain the following matrix equation

$$\begin{pmatrix} 0 \\ T_D \\ T(\mathbf{r}_3) \\ T(\mathbf{r}_4) \\ T(\mathbf{r}_5) \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x}\psi_{11} & \frac{\partial}{\partial x}\psi_{12} & \cdots & \frac{\partial}{\partial x}\psi_{15} \\ \psi_{21} & \psi_{22} & \cdots & \psi_{25} \\ \psi_{31} & \psi_{32} & \cdots & \psi_{35} \\ \psi_{41} & \psi_{42} & \cdots & \psi_{45} \\ \psi_{51} & \psi_{52} & \cdots & \psi_{55} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \vdots \\ \alpha_5 \end{pmatrix}. \tag{6}$$

Note that $T(\mathbf{r}_1)$ of the focal node l is not taken into account in determining the collocation coefficients α_k's, because only the derivative of T is specified by the boundary condition. The temperature of l should be updated by calculating

$$T(\mathbf{r}_1) = \sum_{k=1}^{5} \psi_k(\mathbf{r}_1)\alpha_k, \tag{7}$$

after solving Eq. (6) for α_k's. Repeat this procedure for all the N_Γ sample nodes on Γ.

6. Go back to Step 4 for the next time step.

As mentioned in Step 2, one should be careful in determining ω if the focal node l belongs to Γ. In Fig. 2, we construct ω by choosing the $K = 5$ nearest neighbors of l. Suppose that we impose the Neumann boundary condition on this Γ. The matrix equation to solve is obtained as

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ T(\mathbf{r}_4) \\ T(\mathbf{r}_5) \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x}\psi_{11} & \frac{\partial}{\partial x}\psi_{12} & \cdots & \frac{\partial}{\partial x}\psi_{15} \\ \frac{\partial}{\partial x}\psi_{21} & \frac{\partial}{\partial x}\psi_{22} & \cdots & \frac{\partial}{\partial x}\psi_{25} \\ \frac{\partial}{\partial x}\psi_{31} & \frac{\partial}{\partial x}\psi_{32} & \cdots & \frac{\partial}{\partial x}\psi_{35} \\ \psi_{41} & \psi_{42} & \cdots & \psi_{45} \\ \psi_{51} & \psi_{52} & \cdots & \psi_{55} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{pmatrix}. \tag{8}$$

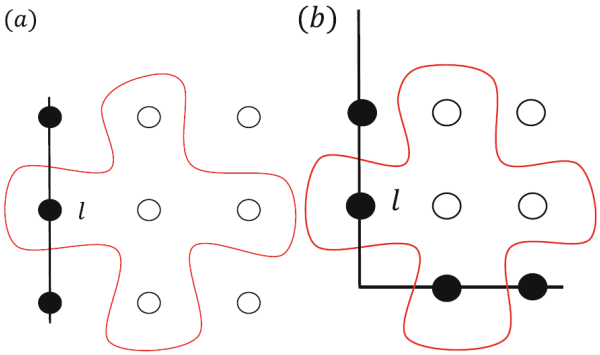

Fig. 3. (Color online) Examples of ω to make the collocation matrix non-singular. The lines are boundaries, and the empty and filled circles represent nodes in Ω and those in Γ, respectively. (a) The focal node l is the only one on the boundary inside ω. (b) The domain of influence ω is located on a corner and contains another boundary node than l.

Note that $\frac{\partial}{\partial x}\psi_{nk}$ is identically zero for $1 \le n \le 3$ and $1 \le k \le 3$, because each ψ_{nk} is a RBF. Therefore, we have only two degrees of freedom, α_4 and α_5, to make three different derivatives vanish. In other words, the matrix is singular. The problem can be avoided by defining ω in a different way so that the number of domain nodes is greater than or equal to that of boundary nodes as shown in Figs. 3(a) and 3(b) (See, *e.g.*, Ref. [11]).

III. METHOD OF IMAGES

The method in the previous section treats a node differently depending on whether it belongs to Ω or Γ. That is, the present value of T on the node does not appear in the collocation matrix when it is subject to a boundary condition specified by the derivative of T. The reason is that one has K unknowns, which implies that the number of equations cannot be greater than K, whereas the node on Γ introduces two equations, one for T and the other for its derivative. The situation could be worse if the node was on a corner so that it should satisfy two or more boundary conditions at the same time.

When we solve the Laplace equation in electrostatics, the boundary conditions can be handled by the method of images [9]. Numerically, the images can be simulated by introducing extra nodes outside Ω: They provide more unknowns, but we do not have to consider neither T nor its derivative on these nodes. The method would work only approximately, because the RBF in Eq. (3) is not an exact solution for Eq. (1). Once again, there is no reason to assume such a regular grid for the image nodes as in Fig. 4. They do not even have to be put outside Ω, as long as the collocation matrix is non-singular.

To illustrate how our method works, Fig. 4(a) shows the domain of influence which led to a singular colloca-

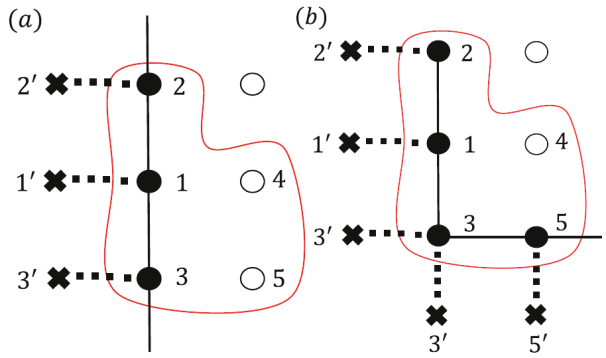

Fig. 4. (Color online) Examples of ω with image nodes, represented by the crosses and primed indices. The lines are boundaries, and the empty and filled circles represent nodes in Ω and those in Γ, respectively.

tion matrix in the previous section. This time, however, every boundary node i is accompanied by an image i' $(i = 1, 2, 3)$. Under the Neumann boundary condition, the collocation equation is written as

$$\begin{pmatrix} T(\mathbf{r}_1) \\ T(\mathbf{r}_2) \\ T(\mathbf{r}_3) \\ T(\mathbf{r}_4) \\ T(\mathbf{r}_5) \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \psi_{11} & \cdots & \psi_{15} & \psi_{11'} & \psi_{12'} & \psi_{13'} \\ \psi_{21} & \cdots & \psi_{25} & \psi_{21'} & \psi_{22'} & \psi_{23'} \\ \psi_{31} & \cdots & \psi_{35} & \psi_{31'} & \psi_{32'} & \psi_{33'} \\ \psi_{41} & \cdots & \psi_{45} & \psi_{41'} & \psi_{42'} & \psi_{43'} \\ \psi_{51} & \cdots & \psi_{55} & \psi_{51'} & \psi_{52'} & \psi_{53'} \\ \frac{\partial}{\partial x}\psi_{11} & \cdots & \frac{\partial}{\partial x}\psi_{15} & \frac{\partial}{\partial x}\psi_{11'} & \frac{\partial}{\partial x}\psi_{12'} & \frac{\partial}{\partial x}\psi_{13'} \\ \frac{\partial}{\partial x}\psi_{21} & \cdots & \frac{\partial}{\partial x}\psi_{25} & \frac{\partial}{\partial x}\psi_{21'} & \frac{\partial}{\partial x}\psi_{22'} & \frac{\partial}{\partial x}\psi_{23'} \\ \frac{\partial}{\partial x}\psi_{31} & \cdots & \frac{\partial}{\partial x}\psi_{35} & \frac{\partial}{\partial x}\psi_{31'} & \frac{\partial}{\partial x}\psi_{32'} & \frac{\partial}{\partial x}\psi_{33'} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_{1'} \\ \alpha_{2'} \\ \alpha_{3'} \end{pmatrix}. \tag{9}$$

Even if a node is located on a corner and thus subject to two different boundary conditions at the same time [Fig. 4(b)], we can readily write down a 10×10 collocation matrix which is non-singular. The insertion of such a corner node is important in reducing numerical error, because a well-known problem of the collocation method is that the result is the most inaccurate near boundaries [2].

Note that Eq. (9) takes care of both T and its derivative on an equal footing. Formally, we may consider images for *every* sample node, even if it belongs to Ω, with setting their contributions to be trivially zero. In this way, we merge Steps 4 and 5 in the previous section and treat all the sample nodes with a single step.

IV. BENCHMARK TEST RESULTS

1. First Test: Boundary Value Problem

As in Ref. [8], we use the NAFEMS benchmark test No. 10 [12]: We consider a rectangular domain $\Omega = (0, L_x) \times (0, L_y)$ with $L_x = 0.6$ m and $L_y = 1.0$ m. The material properties are specified by $k = 52$ W·m^{-1}·°C^{-1}, $c = 460$ J·kg^{-1}·°C^{-1}, and $\rho = 7850$ kg·m^{-3}. The temperature is fixed to $T_D = 100$ °C of the lower boundary at $y = 0$. The left boundary at $x = 0$ is thermally insulated so that the proper choice is the Neumann boundary condition with $\partial T/\partial x|_{x=0} = 0$ °C·m^{-1}. On the other two boundaries, we have heat convection to $T_{\rm ref} = 0$ °C with a convective heat transfer coefficient $h = 750$ W·m^{-2}·°C^{-1}. It is expressed as a Robin boundary condition [Eq. (2)] with $R \equiv -h/k$. Under these boundary conditions, the analytic solution of the Laplace equation for $T(\mathbf{r})$ with $\mathbf{r} \equiv (x, y)$ is given as

$$T_{\rm ana}(\mathbf{r}) = \sum_{n=1}^{\infty} \frac{-2T_D R\cos(\beta_n x)\{\beta_n \cos[\beta_n(L_y - y)] - R\sinh[\beta_n(L_y - y)]\}}{\cos(\beta_n L_x)[\beta_n \cosh(\beta_n L_y) - R\sinh(\beta_n L_y)][L_x(R^2 + \beta_n^2) - R]}, \tag{10}$$

where β_n is the nth positive root of the following equation

$$\beta\tan(\beta L_x) + R = 0. \tag{11}$$

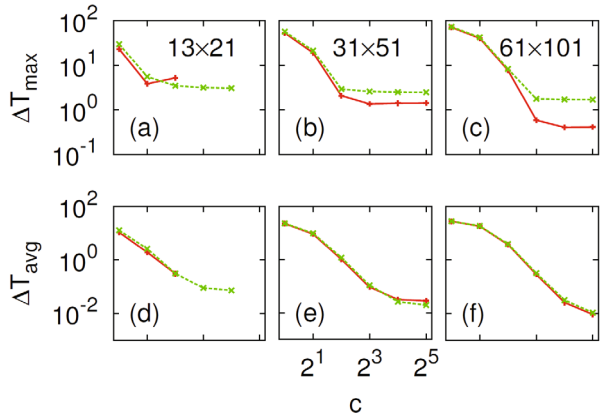

Fig. 5. (Color online) The solid red lines represent the results in Tables 1, 2, and 3, obtained with $K = 5$. The dotted green lines are taken from Ref. [8] for comparison. The upper and lower rows show $\Delta T_{\max}$ and ΔT_{avg}, respectively, and the columns mean different node arrangements from 13 × 21 to 61 × 101. The horizontal axis shows the values of $c = 1, 2, 4, \ldots, 32$. A missing data point means that the solution diverges.

Table 1. First benchmark test result of the image method with domain size $K = 5$ and 13 × 21 nodes. The last two columns show the position of the node with the maximum absolute error $\Delta T_{\max}$. The method becomes unstable for $c \geq 8$.

c	ΔT_{avg} [°C]	$\Delta T_{\max}$ [°C]	$x_{\max}$ [m]	$y_{\max}$ [m]
1	10.7990	23.2401	0.15	0.30
2	1.9253	3.8687	0.15	0.35
4	0.3081	5.2266	0.60	0.05

To check numerical performance, we are concerned with two quantities. One is the maximum absolute deviation of our numerical solution T from the analytic solution T_{ana},

$$\Delta T_{\max} = \max \left| T_{\text{ana}}(\mathbf{r}_n) - T(\mathbf{r}_n) \right|, \tag{12}$$

and the other is the average absolute deviation

$$\Delta T_{\text{avg}} = \frac{1}{N} \sum_{n=1}^{N} \left| T_{\text{ana}}(\mathbf{r}_n) - T(\mathbf{r}_n) \right|, \tag{13}$$

where $\mathbf{r}_n$ denotes the position of the node indexed as n. On the other hand, we can try a quick check by measuring the temperature at a reference point $\mathbf{r}_{\text{NAFEMS}}$ with $x_{\text{NAFEMS}} = 0.6$ m and $y_{\text{NAFEMS}} = 0.2$ m, whose analytic value is $T_{\text{NAFEMS}} \approx 18.2538$ °C according to Eq. (10).

Figure 5 summarizes our main results. It is a graphical representation of the numerical data tabulated in Tables 1, 2, and 3. Note that the results are only for $K = 5$ because our method is unstable for $K = 9$, whereas both the cases are available in Ref. [8]. This may be an example of the trade-off between accuracy and stability [2].

Table 2. First benchmark test result of the image method with 31 × 51 nodes.

c	ΔT_{avg} [°C]	$\Delta T_{\max}$ [°C]	$x_{\max}$ [m]	$y_{\max}$ [m]
1	22.5082	53.2592	0.10	0.20
2	9.1417	19.1447	0.06	0.32
4	1.0199	2.0731	0.06	0.36
8	0.0925	1.3545	0.60	0.02
16	0.0314	1.4061	0.60	0.02
32	0.0281	1.4229	0.60	0.02

Table 3. First benchmark test result of the image method with 61 × 101 nodes.

c	ΔT_{avg} [°C]	$\Delta T_{\max}$ [°C]	$x_{\max}$ [m]	$y_{\max}$ [m]
1	27.6561	70.9615	0.09	0.13
2	18.2390	40.5895	0.05	0.26
4	3.7371	7.7168	0.03	0.36
8	0.2880	0.5874	0.03	0.37
16	0.0255	0.4048	0.60	0.01
32	0.0092	0.4138	0.60	0.01

Table 4. First benchmark test result at $\mathbf{r}_{\text{NAFEMS}}$. The second last column shows error from the analytic solution [Eq. (10)], and the last column is taken from Ref. [8] for comparison.

Nodes	K	c	T [°C]	Error from T_{NAFEMS} [°C]	Error in Ref. [8] [°C]
13 × 21	5	4	17.7508	0.5029	0.1075
31 × 51	5	32	18.1846	0.0692	0.0317
61 × 101	5	32	18.2375	0.0162	0.0056

The figure shows that our image method can significantly reduce the maximum absolute error $\Delta T_{\max}$. For example, for the node arrangement of 61 × 101, $\Delta T_{\max}$ is reduced almost by a factor of 4 compared with the results in Ref. [8] [Fig. 5(c)]. It turns out essential to have corner nodes, such as the one indexed as 3 in Fig. 4(b), to reduce $\Delta T_{\max}$. Those corner nodes can be properly handled by using image nodes, when they have to satisfy more than one condition. Without the images, the maximum absolute error would decrease rather slowly as the number of nodes grows [see the dotted green lines in Figs. 5(a) to (c), which depict the results in Ref. [8]].

Although the image method enhances accuracy in terms of this maximum absolute error, it increases numerical instability. For example, when we work with 13 × 21 nodes, our method give diverging results for $c = 8$ (Table 1), whereas the results would converge without the images [8]. In addition, we should note that the average error decreases only slightly [Fig. 5(d) to (f)] and even increases sometimes [see the rightmost points in Fig. 5(e)]. In Table 4, we check deviations from

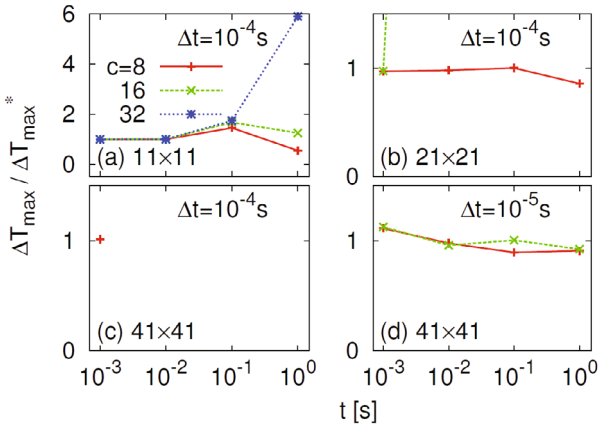

Fig. 6. (Color online) The ratio of our maximum absolute error $\Delta T_{\max}$ with respect to that of Ref. [8], denoted as $\Delta T^*_{\max}$, in the second benchmark test. If the ratio is less than unity, for example, it means that we have a more accurate result than in Ref. [8]. All the results are obtained with $K = 5$. Each panel shows a different combination of the node arrangement and the time step Δt for numerical integration. As in panel (a), a different color means a different value of c, and a missing data point means that the solution diverges.

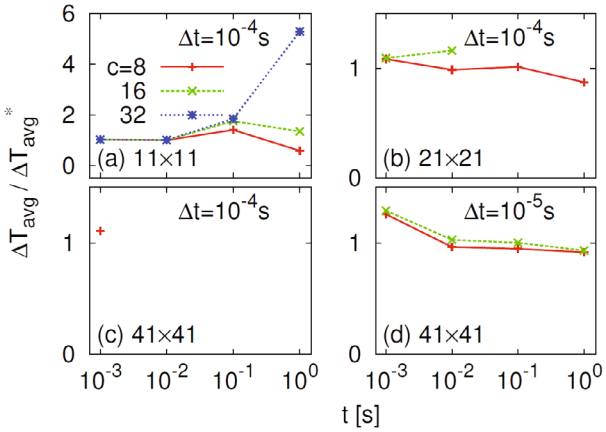

Fig. 7. (Color online) The ratio of our average absolute error $\Delta T_{\rm avg}$ with respect to that of Ref. [8], denoted as $\Delta T^*_{\rm avg}$, in the second benchmark test. If the ratio is less than unity, therefore, it means that we have a more accurate result than in Ref. [8]. The other details are the same as explained in the caption of Fig. 6.

$T_{\rm NAFEMS}$ at $\mathbf{r}_{\rm NAFEMS}$ for different node arrangements. It quickly decreases as the number of nodes increases, but still greater than in Ref. [8].

2. Second Test: Initial Value Problem

Although we are primarily concerned about the boundary value problem, we have also checked the initial value problem addressed in Ref. [8] for completeness. We solve the diffusion equation [Eq. (1)] on a square domain with $L_x = L_y = 1.0$ m. The material properties take unit values, *i.e.*, $\rho = 1$ kg·m^{-3}, $c = 1$ J·kg^{-1}·°C^{-1}, and $k = 1$ W·m^{-1}·°C^{-1}. The boundary conditions are also simplified so that the temperature on the right and upper boundaries is fixed to $T_D = 0$ °C, whereas the other two boundaries are of the Neumann type with zero heat flux. If $T(\mathbf{r}) = 1$ °C at $t = 0$, the analytic solution [13] is given as

$$T_{\rm ana}(\mathbf{r}, t) = T_{\rm ana}(x, t) T_{\rm ana}(y, t), \tag{14}$$

where

$$T_{\rm ana}(q, t) = \frac{4}{\pi} \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} \exp\left[-\frac{k(2n+1)^2\pi^2 t}{4\rho c L_q^2}\right] \times \cos\left[\frac{(2n+1)\pi q}{2L_q}\right], \tag{15}$$

where q means either x or y.

The results are tabulated in Tables 5 to 8, and their graphical representations are given in Figs. 6 and 7. Overall, we get slightly better numerical accuracy than in Ref. [8] when it comes to the largest number of sample nodes and the smallest $\Delta t = 10^{-5}$ s. The price is numerical instability in that the result blows up with $c = 32$ except for the smallest number of sample nodes. This result is not very surprising, however, because the method of images is meant to deal with more complicated boundary-value problems.

Table 5. Second benchmark test result of the image method with $\Delta t = 10^{-4}$ s and 11 × 11 nodes. The size of ω is set to be $K = 5$ for every l.

t [s]	c	$\Delta T_{\rm avg}$ [°C]	$\Delta T_{\max}$ [°C]	$x_{\max}$ [m]	$y_{\max}$ [m]
10^{-3}	8	1.206e-02	1.245e-01	0.900	0.900
10^{-3}	16	1.204e-02	1.243e-01	0.900	0.900
10^{-3}	32	1.204e-02	1.243e-01	0.900	0.900
10^{-2}	8	4.864e-03	2.265e-02	0.700	0.700
10^{-2}	16	4.787e-03	2.231e-02	0.700	0.700
10^{-2}	32	4.769e-03	2.222e-02	0.700	0.700
10^{-1}	8	1.743e-03	5.015e-03	0.000	0.100
10^{-1}	16	1.250e-03	4.330e-03	0.000	0.100
10^{-1}	32	1.148e-03	4.168e-03	0.000	0.100
10^{0}	8	4.071e-05	9.704e-05	0.000	0.000
10^{0}	16	1.222e-05	3.498e-05	0.100	0.000
10^{0}	32	2.477e-05	6.604e-05	0.100	0.000

V. DISCUSSION AND SUMMARY

In summary, we have modified the local RBF collocation method by adding an additional set of nodes in the same spirit of PDE collocation on the boundary in Ref. [10]. This method makes it possible to take into ac-

Table 6. Second benchmark test result of the image method with $\Delta t = 10^{-4}$ s and 21 $\times$ 21 nodes. The size of ω is set to be $K = 5$ for every l. For $c = 16$ and 32, the solution diverges as time goes by.

t [s]	c	$\Delta T_{\rm avg}$ [°C]	$\Delta T_{\rm max}$ [°C]	$x_{\rm max}$ [m]	$y_{\rm max}$ [m]
10^{-3}	8	4.984e-03	4.283e-02	0.900	0.900
10^{-3}	16	4.949e-03	4.257e-02	0.900	0.900
10^{-3}	32	5.254e-03	1.356e-01	0.050	0.000
10^{-2}	8	1.553e-03	6.983e-03	0.750	0.750
10^{-2}	16	1.545e-03	7.971e-02	0.050	0.000
10^{-1}	8	1.258e-03	2.390e-03	0.250	0.100
10^{0}	8	9.355e-05	2.310e-04	0.000	0.000

Table 7. Second benchmark test result of the image method with $\Delta t = 10^{-4}$ s and 41 $\times$ 41 nodes. The size of ω is set to be $K = 5$ for every l. The result diverges for $c = 16$ and 32, and it is the case even for $c = 8$ when $t \gtrsim 10^{-2}$.

t [s]	c	$\Delta T_{\rm avg}$ [°C]	$\Delta T_{\rm max}$ [°C]	$x_{\rm max}$ [m]	$y_{\rm max}$ [m]
10^{-3}	8	1.993e-03	2.428e-02	0.950	0.950

Table 8. Second benchmark test result of the image method with $\Delta t = 10^{-5}$ s and 41 $\times$ 41 nodes. The size of ω is set to be $K = 5$ for every l. We see diverging results for $c = 32$.

t [s]	c	$\Delta T_{\rm avg}$ [°C]	$\Delta T_{\rm max}$ [°C]	$x_{\rm max}$ [m]	$y_{\rm max}$ [m]
10^{-3}	8	1.556e-03	1.745e-02	0.925	0.925
10^{-3}	16	1.519e-03	1.688e-02	0.925	0.925
10^{-2}	8	7.795e-04	2.577e-03	0.775	0.775
10^{-2}	16	3.932e-04	1.696e-03	0.750	0.750
10^{-1}	8	1.865e-03	3.448e-03	0.225	0.100
10^{-1}	16	3.160e-04	6.349e-04	0.125	0.025
10^{0}	8	1.892e-04	4.676e-04	0.000	0.000
10^{0}	16	2.441e-05	6.034e-05	0.000	0.000

count every piece of available information on the boundaries. That is, our collocation matrix can describe both the functional value T as well as its spatial derivatives on every boundary node even if the node is subject to two or more boundary conditions. This small modification is able to reduce the maximum error $\Delta T_{\rm max}$ relative to the analytic solution almost by a factor of 4 in the first benchmark test for a boundary value problem [see Fig. 5(c)]. It should be noted that the collocation at the boundaries makes the numerical integration more unstable. It is therefore desirable to use more sample nodes and smaller time steps for convergence, and one could think of implementing an implicit scheme such as the Crank-Nicholson method rather than our simple Euler scheme. We do not pursue this direction because our purpose is to make a direct comparison with Ref. [8]. Overall, if a boundary value problem is given with Robin boundary conditions, we can recommend including collocation at the boundaries: Combined with the local RBF collocation method, the additional amount of effort is small whereas the reduction of the maximum error is significant, as long as the result is convergent with a sufficiently large number of sample nodes.

ACKNOWLEDGMENTS

This work was supported by a research grant of Pukyong National University (2015).

REFERENCES

[1] E. J. Kansa, Comput. Math. Appl. **19**, 147 (1990).
[2] G. E. Fasshauer, *Meshfree Approximation Methods with MATLAB* (World Scientific, Singapore, 2007).
[3] Y. C. Hon and X. Z. Mao, Appl. Math. Comput. **95**, 37 (1998).
[4] Y-C. Hon, K. F. Cheung, X-Z. Mao and E. J. Kansa, J. Hydraul. Eng. **125**, 524 (1999).
[5] E. Larsson and B. Fornberg, Comp. Math. Appl. **46**, 891 (2003).
[6] J. Perko, C. S. Chen and B. Šarler, WIT Transactions on Modelling and Simulation **29**, 111 (2001).
[7] W. Chen, L. Ye and H. Sun, Comput. Math. Appl. **59**, 1614 (2010).
[8] Šarler and R. Vertnik, Comput. Math. Appl. **51**, 1269 (2006).
[9] J. D. Jackson, *Classical Electrodynamics*, 3rd ed. (John Wiley & Sons, New York, 1999).
[10] A. I. Fedoseyev, M. J. Friedman and E. J. Kansa, Comput. Math. Appl. **43**, 439 (2002).
[11] Y-C. Hon, B. Šarler and D. Yun, Eng. Anal. Bound. Elem. **57**, 2 (2015).
[12] A. D. Cameron, J. A. Casey and G. B. Simpson, *Benchmark Test For Thermal Analysis* (National Agency for Finite Element Methods & Standards, Glasgow, 1986).
[13] H. S. Carslaw and J. C. Jaeger, *Conduction of Heat in Solids*, 2nd ed. (Clarendon Press, Oxford, 1959).